

IN THE CLAIMS:

1. (Currently Amended) In a computer-operated system for programming a manufacturing system, wherein a version of a program is downloaded to the manufacturing system and is associated with a version designator, a method for managing revisions to versions of the program, the method comprising the steps of:

running the program on the manufacturing system, the program including subsets organized with respect to one another in a hierarchical fashion, the subsets comprising a top-level subset and a plurality of lower-level subsets related hierarchically to the top-level subsets and to each other, each of the subsets identifying a main object and a sub-object, a version designator being associated with each of the subsets, each version designator including a first version identifier associated with the main object and a second version identifier associated the sub-object; and

if the program, when run on the manufacturing system, performs according to a preselected criterion, revising the first version designator identifier of a certain lower-level subset when the certain lower-level subset is modified for the program; and

if the modification of the certain lower-level subset affects other subsets, revising the second version identifier of the other affected subsets.

2. (Currently Amended) The method according to claim 1, wherein the version designator comprises along with the first and second version identifiers, a version label, ~~version-identifier~~, and a status identifier.

3. Canceled

4. (Currently Amended) The method according to claim 1, wherein each of the first and second the-version identifiers comprises a number and the step of ~~designating a revised~~ revising the first and second version identifiers ~~designator for the program~~ comprises the step of incrementing the version identifier.

5. (Original) The method according to claim 2, wherein the status identifier comprises a validation indicator.

(Currently Amended) The method according to claim 5, wherein the manufacturing system comprises at least one line, ~~the preselected criterion comprises a check whether the program runs satisfactorily on the line, the method further including and wherein the step of designating a revised status identifier for the program comprises~~ activating the validation indicator after the program runs satisfactorily on the line.

6. (Original) The method according to claim 6, wherein the validation indicator is checked as a condition to running the program on the line without manual intervention.

7. (Original) The method according to claim 2, wherein the status identifier comprises a release indicator.

8. (Currently Amended) The method according to claim 8, wherein the manufacturing system comprises a plurality of lines, ~~the preselected criterion comprises a determination that the program runs satisfactorily on the lines, and the release indicator identifies the program as released for use on the plurality of lines~~ after the program runs satisfactorily on the lines.

9. (Original) The method according to claim 8, wherein the release indicator is checked as a condition to running the program on any lines of the manufacturing system without manual intervention.

10. (Original) The method according to claim 1, wherein the manufacturing system comprises an electronics assembly system.

11. (Currently Amended) A computer readable medium having a data structure stored thereon for use in identifying programs for computer-controlled manufacturing systems, wherein the programs comprise subsets organized with respect to one

another in a hierarchical fashion, the subsets comprising a top-level subset and a plurality of lower-level subsets related hierarchically to the top-level subsets and to each other, the data structure of the computer readable medium comprising:

a first portion for indicating a revision to the top-level subset of a program; ~~and~~
a second portion for indicating a revision to any of the lower-level subsets of the program; and
a label indicating a status of the programs.

12. (Currently Amended) The medium ~~computer data structure~~ according to claim 12, wherein either of the first and second portions comprise a numeral that is incremented to identify the revision, and wherein the label is a letter that identifies the status as one of released and validated.

13. (Currently Amended) The medium ~~computer data structure~~ according to claim 12, wherein the first and second portions are positioned adjacent one another.

14. (Currently Amended) The medium ~~computer data structure~~ according to claim 13, wherein the first and second portions are positioned on either side of a decimal point.

15. (Currently Amended) A method for completing a data structure, stored on a computer readable medium, for use in identifying programs for computer-controlled manufacturing systems, wherein the programs comprise subsets organized with respect to one another in a hierarchical fashion, the subsets comprising a top-level subset and a plurality of lower-level subsets related hierarchically to the top-level subsets and to each other, the method comprising the steps of:

in a first portion of the data structure for indicating a revision to the top-level subset of a program, storing, on the computer readable medium, ~~inserting~~ a first symbol indicating that such a revision has been made; ~~and~~

in a second portion of the data structure for indicating a revision to any of the plurality of lower-level subsets of the program, storing, on the computer readable medium, inserting a second symbol indicating that such a revision has been made, and
in a third portion of the data structure for indicating a status of the program,
storing, on the computer readable medium, a third symbol indicating the status to be one of released and validated.

16. (Currently Amended) The method according to claim 16, wherein the first portion comprises an identifier for the top-level subset, and the second portion comprises an identifier for any of the plurality of the lower-level subsets.

17. (Original) The method according to claim 16, wherein the top-level subset comprises a main object.

18. (Original) The method according to claim 18, wherein the main object comprises a representation of a circuit board.

19. (Original) The method according to claim 18, wherein the lower-level subsets comprise sub-objects relative to the main object.

20. (Original) The method according to claim 20, wherein the main object comprises a representation of a circuit board and the sub-objects comprise representations of components to be place on the circuit board.

21. (Currently Amended) A ~~computer-implemented~~ method for managing revisions to a program used in the control of a manufacturing system, the method comprising the steps of:

running the program on the manufacturing system,

indicating to a user identifying that a revision has been made to the program;

based on certain criteria, approving the revised program by the user, and

based on an authority level of the user, designating a status of the revised program as one of released and validated.

~~identifying whether the program, as revised, satisfies a preselected criterion;
if the program, as revised, satisfies the preselected criterion, automatically selecting a
version designator according to a preselected scheme; and
automatically associating the selected version indicator with the program; and
code.~~

22. (Original) The method according to claim 22, wherein the manufacturing system comprises an electronics assembly system.

23. Canceled

24. Canceled

25. Canceled

26. Canceled

27. Canceled